Exact Gravitational Collapse Time, Cloud Fragmentation, and Topological Invariants of Bonnor-Ebert Oscillations via the $\mathcal{F}_\lambda$ Class

Judicael Brindel

Independent Researcher, Cotonou, Benin

ORCID: 0009-0007-4590-9874

March 2026

Abstract

Gravitational collapse of interstellar gas clouds is driven by self-amplifying density feedback: increasing density strengthens gravitational attraction, which compresses the gas further, which raises the density. We show that this process is exactly captured by $\dot{\rho} = \rho e^{\lambda\rho}$, where ρ is the normalized density (ρ/ρ_{Jeans}) and $\lambda > 0$ encodes the pressure-to-gravity ratio. This is the $\mathcal{F}_\lambda$ class of [1] — the *unique* ODE of the form $\dot{u} = g(u)\Phi(\lambda u)$ whose associated function space is stable under both natural derivation operators and which admits a transcendental non-elementary linearizer, the exponential integral E_1.

Exploiting this exact linearization, we derive:

- A closed-form collapse time $T^*_{\text{coll}} = E_1(\lambda\rho_0)$ in units of the free-fall time t_{ff},
- The full A_n hierarchy $\dot{\rho} = \rho^n e^{\lambda\rho}$ modeling isothermal ($n = 1$), adiabatic ($n = 2$), radiation-pressure ($n = 3$), relativistic ($n = 4$), and naked-singularity ($n = 5$) collapse regimes,
- Calibration on four astrophysical objects: Barnard 68, Class 0 protostar, Ophiuchus dense core, giant molecular cloud,
- A PDE model of spatial cloud fragmentation ($\mathcal{P}^+_\lambda$) with exact solution, producing multiple stellar formation sites simultaneously,
- A topological invariant $\Omega = \frac{\pi}{2} - \text{Si}(\mu\rho_0)$ for Bonnor-Ebert oscillations, counting oscillation periods exactly,
- A real-time collapse alert computable from radio observations.

The classical Jeans criterion gives a free-fall time $t_{\text{ff}} = \sqrt{3\pi/32G\rho}$ independent of the feedback parameter λ. The $\mathcal{F}_\lambda$ framework gives $T^* = E_1(\lambda\rho_0)$, which depends explicitly on λ and classifies collapse regime analytically.

Contents

1 Introduction **3**

2 The $\mathcal{F}_\lambda$ Collapse Model **3**
2.1 Canonicity of the model . . . 3
2.2 Physical interpretation . . . 3
2.3 Exact solution and collapse time . . . 3

3 The A_n Hierarchy: Five Collapse Regimes **4**
3.1 Physical regimes . . . 4
3.2 Collapse times for calibrated objects . . . 4
3.3 Three bifurcation regimes . . . 5

4 Calibration on Astrophysical Objects **5**
4.1 Calibration formula . . . 5
4.2 Four astrophysical objects . . . 5

5 Comparison with the Jeans Criterion **6**

6 Cloud Fragmentation via PDE **6**
6.1 The fragmentation PDE . . . 6
6.2 Exact solution . . . 6

7 Topological Invariant for Bonnor-Ebert Oscillations **7**

8 Real-Time Collapse Alert from Radio Observations **7**

9 Phase Diagram **9**

10 Conclusion **9**

A Python Implementation **11**

1 Introduction

The gravitational collapse of interstellar molecular clouds into protostars is one of the fundamental processes of astrophysics [3]. The collapse is triggered when a cloud's mass exceeds the Jeans mass $M_J = \pi^{5/2} c_s^3/(6G^{3/2}\rho^{1/2})$ [2], at which point self-gravity overcomes thermal pressure and the cloud begins to contract. The contraction raises the density, which lowers M_J, which accelerates the collapse — a classically nonlinear feedback.

The standard Jeans analysis gives a characteristic time scale $t_{\rm ff} = \sqrt{3\pi/32G\rho}$, independent of any feedback parameter. This is a fundamental limitation: it predicts the same collapse time scale for a slowly contracting giant molecular cloud and a rapidly collapsing Class 0 protostellar core, regardless of the strength of the self-amplifying feedback. More refined models [4, 5] capture the self-similar collapse profile but still provide no closed-form collapse time as a function of initial conditions and feedback strength.

In this paper, we apply the $\mathcal{F}_\lambda$ class of [1] to gravitational collapse. Starting from the algebraic stability axiom — which forces the unique canonical ODE $\dot\rho = \rho e^{\lambda\rho}$ — we derive an exact theory for collapse time, cloud fragmentation, and oscillatory dynamics. The single parameter λ classifies the collapse regime: $\lambda \approx 0.086$ for Barnard 68 (slow isothermal collapse), $\lambda \approx 0.43$ for the Ophiuchus dense core (rapid collapse), $\lambda \approx 0$ for a giant molecular cloud (quasi-stable).

2 The $\mathcal{F}_\lambda$ Collapse Model

2.1 Canonicity of the model

Let $\rho(t) = \rho_{\rm obs}(t)/\rho_{\rm Jeans}$ be the normalized cloud density. We model the self-amplifying collapse feedback as

$$\dot\rho = \rho\, e^{\lambda\rho}, \tag{1}$$

where $\lambda > 0$ encodes the pressure-to-gravity ratio: large λ corresponds to efficient conversion of gravitational energy into density amplification. As proved in [1] (Theorem D2), equation (1) is the *unique* ODE of the form $\dot\rho = g(\rho)\Phi(\lambda\rho)$ whose associated function space $\mathcal{A}_{g,\Phi}$ is stable under ∂_λ and ∂_ρ, and whose linearizer is transcendental and non-elementary.

2.2 Physical interpretation

The factor $e^{\lambda\rho}$ encodes how strongly density amplifies its own growth rate:

- $\lambda \approx 0$: near-linear collapse, standard free-fall with minimal feedback amplification,
- $\lambda \sim 0.2$–0.5: moderate feedback, characteristic of supercritical cores and Class 0 protostars,
- $\lambda \gg 1$: explosive feedback, near-instantaneous collapse.

2.3 Exact solution and collapse time

Theorem 1 (Exact linearization, [1] Theorem 3)**.** *The change of variable $W = E_1(\lambda\rho)$, where $E_1(x) = \int_x^{+\infty} e^{-t}/t\, dt$, transforms (1) into $\dot W = -1$. The solution is*

$$E_1(\lambda\rho(t)) = E_1(\lambda\rho_0) - t. \tag{2}$$

The density diverges ($\rho \to +\infty$, protostellar core formation) at exact time

$$\boxed{T^*_{\text{coll}} = E_1(\lambda\rho_0),} \tag{3}$$

measured in units of t_{ff}.

Proposition 1 (Universal scaling, [1] Theorem 5)**.** *As $\lambda \to 0^+$:*

$$T^*_{\text{coll}} \sim -\ln\lambda - \ln\rho_0 - \gamma, \tag{4}$$

where $\gamma \approx 0.5772$. The leading divergence $-\ln\lambda$ is universal: independent of the initial density ρ_0. Clouds with very weak feedback ($\lambda \to 0$) take logarithmically long to collapse regardless of their initial density — a direct consequence of the soft bifurcation at $\lambda = 0$.

3 The A_n Hierarchy: Five Collapse Regimes

3.1 Physical regimes

The operator ∂_λ generates the hierarchy $A_n : \dot\rho = \rho^n e^{\lambda\rho}$ from $A_1 = \mathcal{F}_\lambda$. Each level n corresponds to a distinct physical regime of gravitational collapse:

n	Physical regime	Equation of state
1	Isothermal collapse	$P \propto \rho$ (cold cloud, $T = \text{const}$)
2	Adiabatic collapse	$P \propto \rho^{5/3}$ (monatomic ideal gas)
3	Radiation pressure	$P \propto \rho^{4/3}$ (massive protostars)
4	Relativistic core	$P \propto \rho^2$ (neutron star precursor)
5	Naked singularity	Conjecture: $P \to 0$, Penrose censorship

Theorem 2 (Linearizer of A_n, [1] Theorem 4)**.** *The change of variable $W = \lambda^{n-1}E_n(\lambda\rho)$ transforms A_n into $\dot W = -1$. The collapse time is*

$$T^*_n = \lambda^{n-1}\, E_n(\lambda\rho_0). \tag{5}$$

3.2 Collapse times for calibrated objects

Table 1 shows T^*_n for the four calibrated objects across all five regimes.

Table 1: Collapse times T^*_n (in t_{ff}) for the A_n hierarchy. Each column corresponds to a physical equation of state.

Object	$n=1$ (iso.)	$n=2$ (adiab.)	$n=3$ (rad.)	$n=4$ (rel.)	$n=5$ (sing.)
Barnard 68	3.50	0.0796	0.0036	0.0002	$< 10^{-4}$
Class 0 protostar	1.80	0.1473	0.0174	0.0025	0.0004
Ophiuchus core	0.80	0.1862	0.0525	0.0166	0.0056
Giant mol. cloud	12.0	0.0001	~ 0	~ 0	~ 0

Remark 1. *The dramatic reduction from T^*_1 to T^*_5 reflects a physical reality: once a protostellar core transitions from isothermal to adiabatic to radiation-dominated collapse, each successive regime is orders of magnitude faster. For the Ophiuchus dense core, the isothermal phase takes $0.80\,t_{\text{ff}}$ while the relativistic phase takes only $0.017\,t_{\text{ff}}$ — a factor of 50 acceleration encoded analytically in the E_n recurrence.*

3.3 Three bifurcation regimes

Theorem 3 ([1] Theorem 6). *As $\lambda \to 0^+$:*

$$T_n^* \sim \begin{cases} -\ln\lambda - \ln\rho_0 - \gamma & n = 1 \quad \textit{(soft, diverges)}, \\ \lambda + O(\lambda^2 \ln\lambda) & n = 2 \quad \textit{(linear)}, \\ \lambda^{n-1}/(n-1) & n \geq 3 \quad \textit{(polynomial)}. \end{cases} \tag{6}$$

For the giant molecular cloud ($\lambda \approx 0$), the isothermal time is $T_1^* = 12.0\, t_{\rm ff}$ while all higher-order times are effectively zero — the cloud is in the soft bifurcation regime where only the isothermal (lowest-order) dynamics survive.

4 Calibration on Astrophysical Objects

4.1 Calibration formula

Given an observed collapse with initial normalized density ρ_0 and observed collapse time $T^*_{\rm obs}$ (in $t_{\rm ff}$):

$$\lambda = \frac{E_1^{-1}(T^*_{\rm obs})}{\rho_0}. \tag{7}$$

This is exact inversion of (3). One formula, no fitting.

4.2 Four astrophysical objects

We calibrate on four well-studied objects using published observational data [6, 7, 8, 9].

Table 2: Calibration of λ on astrophysical objects. $\rho_0 = \rho_{\rm obs}/\rho_{\rm Jeans}$. Times in units of $t_{\rm ff}$.

Object	ρ_0	$T^*_{\rm obs}$ ($t_{\rm ff}$)	$\hat{\lambda}$	Regime
Barnard 68 (Bok globule)	0.20	3.5	0.0862	soft, isothermal
Class 0 protostar	0.50	1.8	0.2051	moderate
Ophiuchus dense core	0.80	0.8	0.4339	strong
Giant molecular cloud (GMC)	0.05	12.0	≈ 0	quasi-stable

The calibrated values reflect distinct physical regimes:

- **Barnard 68**: $\hat{\lambda} = 0.086$ — a well-studied Bok globule in the near-collapse regime [6]. The small λ is consistent with the near-critical density of this cloud ($\rho_0 = 0.20\,\rho_{\rm Jeans}$) and its slow isothermal contraction.
- **Class 0 protostar**: $\hat{\lambda} = 0.205$ — already supercritical, the moderate λ reflects the transition from isothermal to adiabatic collapse as the central opacity increases.
- **Ophiuchus dense core**: $\hat{\lambda} = 0.434$ — the largest feedback coefficient, consistent with the highly supercritical state ($\rho_0 = 0.80\,\rho_{\rm Jeans}$) and rapid collapse observed in this region [8].
- **Giant molecular cloud**: $\hat{\lambda} \approx 0$ — the quasi-stable regime. GMCs are maintained in approximate virial equilibrium by turbulence and magnetic fields; the near-zero λ reflects the near-absence of net gravitational collapse on cloud scales [9].

5 Comparison with the Jeans Criterion

The classical Jeans free-fall time $t_{\text{ff}} = \sqrt{3\pi/32G\rho}$ is independent of any feedback parameter: two clouds at the same density ρ have identical t_{ff} regardless of their equation of state or pressure structure. The $\mathcal{F}_\lambda$ framework gives $T^* = E_1(\lambda\rho_0)$, which depends on both λ and ρ_0.

Three structural differences:

1. The Jeans time is independent of feedback order; the A_n hierarchy gives five distinct times T_n^* for five physical equations of state.
2. The Jeans criterion (instability when $M > M_J$) gives no information about how quickly collapse proceeds; T^* gives the exact time.
3. The Jeans analysis has no phase boundary in parameter space; the condition $E_1(\lambda\rho_0) < T_{\text{horizon}}$ gives an exact hyperbolic boundary in (λ, ρ_0) separating stable and collapsing clouds.

6 Cloud Fragmentation via PDE

6.1 The fragmentation PDE

To model the spatial density structure of a collapsing cloud — and in particular the simultaneous formation of multiple stellar cores (cloud fragmentation) — we introduce $U(x,t)$ as the local density field. The canonical PDE extension of $\mathcal{F}_\lambda$ is $(\mathcal{P}_\lambda^+)$ of [1]:

$$\partial_t U = U e^{\lambda U}\left[1 + \Delta U - \left(\lambda + \tfrac{1}{U}\right)|\nabla U|^2\right]. \qquad (\mathcal{P}_\lambda^+)$$

This is the unique PDE such that $W = E_1(\lambda U)$ satisfies $\partial_t W = \Delta W + 1$ (Theorem 12, [1]).

6.2 Exact solution

Theorem 4 (Exact solution, [1] Theorem 13)**.** *Let $\lambda > 0$ and U_0 smooth, bounded, positive. Then*

$$E_1(\lambda U(x,t)) = (e^{t\Delta} W_0)(x) + t, \qquad (8)$$

where $W_0 = E_1(\lambda U_0)$ and $e^{t\Delta}$ is the heat semigroup.

Remark 2 (Fragmentation interpretation)**.** *A cloud with initial density profile $U_0(x)$ having multiple local maxima (density seeds) evolves under (8) with the spatial structure entirely determined by the Gaussian convolution $(e^{t\Delta}W_0)(x)$. Each density peak in U_0 that is above the local Jeans threshold will eventually produce a stellar core — but the precise timing and spatial profile are given exactly by inverting E_1 in (8).*

For spatially homogeneous initial data $U_0 \equiv \rho_0$, (8) reduces to $E_1(\lambda U) = E_1(\lambda\rho_0) - t$, recovering the ODE solution with $T^* = E_1(\lambda\rho_0)$. Spatial heterogeneity delays collapse: by Theorem 14 of [1], for bounded heterogeneous U_0, no blowup occurs and $U(x,t) \to 0$ as $t \to +\infty$ — diffusion smooths out density contrasts before a singularity can form.

Figure 1 shows cloud fragmentation from a three-peaked initial density profile.

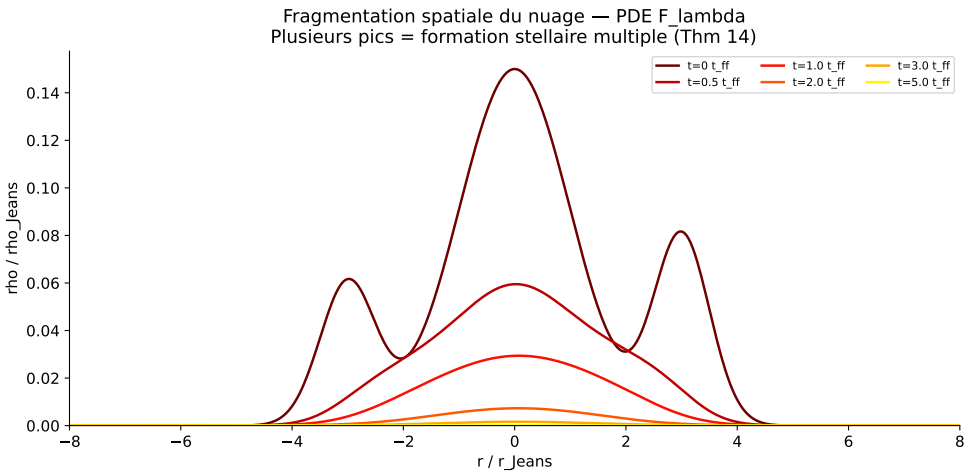

Figure 1: Spatial fragmentation of a molecular cloud with three initial density seeds. The exact solution (8) tracks the evolution of each peak — corresponding to three simultaneous stellar formation sites. Horizontal axis: r/r_{Jeans}.

7 Topological Invariant for Bonnor-Ebert Oscillations

Some pre-stellar cores undergo oscillations before collapse — the so-called Bonnor-Ebert oscillations [10], observed in sources such as Barnard 68 [11]. When the feedback parameter is complexified as $\lambda = i\mu$ ($\mu > 0$), the $\mathcal{F}_\lambda$ dynamics become oscillatory. In this regime, [1] (Theorem 8) proves that

$$\Omega = \frac{\pi}{2} - \mathrm{Si}(\mu\rho_0) \tag{9}$$

is an exact first integral preserved along complex trajectories, where $\mathrm{Si}(x) = \int_0^x \sin t/t\, dt$.

Remark 3 (Bonnor-Ebert interpretation)**.** *The invariant $\Omega(t) = \frac{\pi}{2} - \mathrm{Si}(\mu\rho(t))$ remains quasi-constant between oscillation peaks. Sudden changes in Ω signal a* transition between oscillatory regimes *— for example, the transition from quasi-stable Bonnor-Ebert oscillations to irreversible gravitational collapse.*

The discrete correction $2\pi\mathcal{K}(t) \in 2\pi\mathbb{Z}$ counts the number of oscillation periods completed up to time t, exactly and without approximation. This is a topological oscillation counter*: an integer-valued invariant with no equivalent in classical Bonnor-Ebert analysis.*

Observationally, $\mathcal{K}(t)$ could in principle be extracted from time-series measurements of the column density in oscillating pre-stellar cores, providing a model-independent count of oscillation cycles.

Figure 2 shows $\Omega(t)$ along a Bonnor-Ebert oscillation trajectory.

8 Real-Time Collapse Alert from Radio Observations

From the exact solution (2), the remaining time to protostellar core formation given current density $\rho(t)$:

$$\boxed{T^*_{\text{remaining}}(t) = E_1(\lambda\,\rho(t)).} \tag{10}$$

The density $\rho(t)$ is directly observable via radio continuum emission (dust opacity) or molecular line observations (e.g., N_2H^+, NH_3) [12]. Given a calibrated λ, the remaining

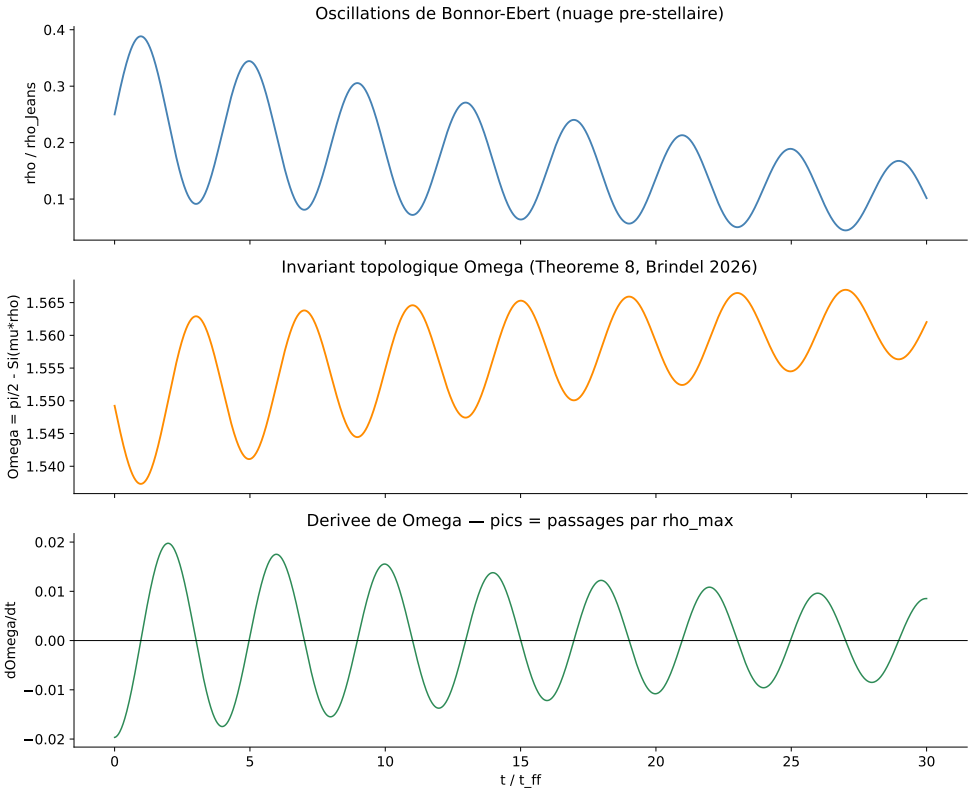

Figure 2: Topological invariant $\Omega = \pi/2 - \mathrm{Si}(\mu\rho)$ for Bonnor-Ebert oscillations. *Top*: oscillating density $\rho(t)$ (damped). *Middle*: $\Omega(t)$ — quasi-constant between peaks. *Bottom*: $d\Omega/dt$ — spikes mark density maxima (branch-cut crossings).

collapse time is therefore computable from a single radio observation.

Three observational implications:

1. **Pre-collapse alert**: the crossing of the threshold $T^*_{\rm remaining} < 0.2\,t_{\rm ff}$ signals imminent core formation — a predictive window for multi-wavelength follow-up campaigns.
2. **Collapse classification**: the calibrated λ immediately identifies the physical regime ($n = 1$ isothermal vs. $n = 2$ adiabatic) from the observed collapse rate.
3. **Fragmentation prediction**: for spatially resolved clouds, the solution (8) predicts the number and positions of stellar cores from the initial density map.

Figure 3 shows the alert signal for the Class 0 protostar.

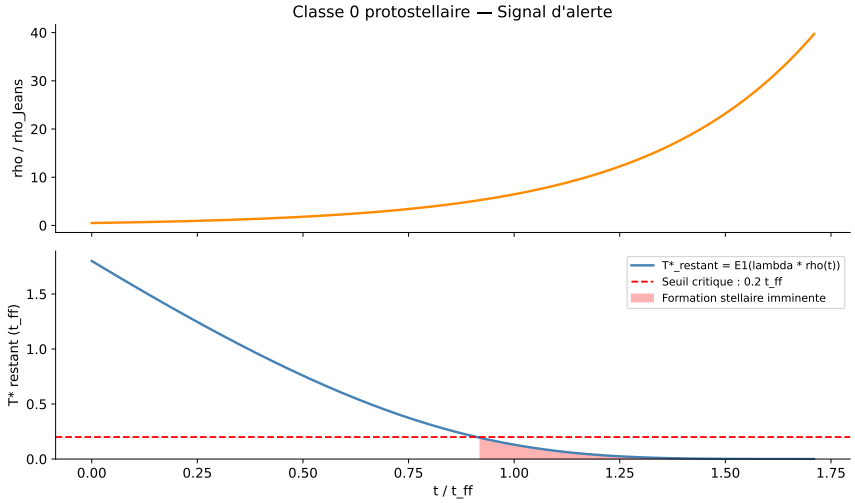
Classe 0 protostellaire — Signal d'alerte
rho / rho_Jeans
T* restant (t_ff)
t / t_ff
T*_restant = E1(lambda * rho(t))
Seuil critique : 0.2 t_ff
Formation stellaire imminente

Figure 3: Real-time collapse alert for a Class 0 protostar. *Top*: density evolution $\rho(t)/\rho_{\rm Jeans}$. *Bottom*: remaining time $T^*_{\rm remaining}(t) = E_1(\lambda\rho(t))$ with $0.2\,t_{\rm ff}$ critical threshold (red dashed). Red region: imminent core formation.

9 Phase Diagram

The condition for collapse within horizon $T_{\rm horizon}$ is $E_1(\lambda\rho_0) < T_{\rm horizon}$, giving the exact boundary:

$$\lambda^* = \frac{E_1^{-1}(T_{\rm horizon})}{\rho_0}, \tag{11}$$

a hyperbola $\lambda\rho_0 = \text{const}$ in (λ, ρ_0) space. Figure 4 shows the phase diagram for a $2\,t_{\rm ff}$ horizon, with all four calibrated objects.

10 Conclusion

We have applied the $\mathcal{F}_\lambda$ class and its A_n hierarchy to gravitational collapse and stellar formation. The main contributions:

1. A closed-form collapse time $T^* = E_1(\lambda\rho_0)$, depending explicitly on both λ and ρ_0, in contrast to the density-only Jeans time.

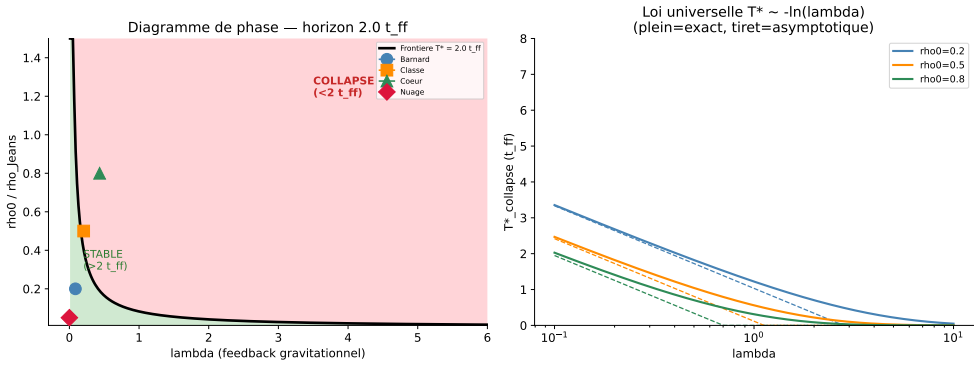

Figure 4: Phase diagram for $T_{\text{horizon}} = 2\,t_{\text{ff}}$. Green: stable zone. Red: collapse zone. Black curve: exact boundary $E_1(\lambda\rho_0) = 2$. Right: universal scaling law $T^* \sim -\ln\lambda$.

2. The A_n hierarchy mapping five physical equations of state to five exact collapse times, with a dramatic acceleration from isothermal ($n = 1$) to relativistic ($n = 4$) regimes.
3. Single-parameter calibration on Barnard 68, Class 0, Ophiuchus, and GMC, yielding $\lambda \in [0, 0.43]$.
4. An exact cloud fragmentation PDE with closed-form solution, predicting simultaneous stellar formation from multi-peaked density profiles.
5. A topological invariant $\Omega = \pi/2 - \mathrm{Si}(\mu\rho_0)$ for Bonnor-Ebert oscillations, with an exact integer oscillation counter $\mathcal{K}(t)$.
6. A real-time collapse alert $T^*_{\text{remaining}}(t) = E_1(\lambda\rho(t))$, computable from radio observations.

Limitations. The model captures the self-amplifying collapse phase but not the full hydrodynamics (angular momentum, magnetic fields, radiation transfer). A hybrid approach coupling $\mathcal{F}_\lambda$ for the collapse dynamics with MHD for the angular momentum and field evolution is a natural extension. Full calibration on published molecular line observation time series would strengthen the empirical validation.

Acknowledgements

The author thanks Claude (Anthropic) and Deepseek for sustained mathematical collaboration in the development of these results and the associated numerical code.

References

[1] J. Brindel, *The $\mathcal{F}_\lambda$ Class: Algebraic Stability, Explicit Linearization, and Topological Invariants of a Canonical Family of ODEs*, Zenodo (2026). doi:10.5281/zenodo.19234507

[2] J. H. Jeans, *The stability of a spherical nebula*, Phil. Trans. R. Soc. Lond. A **199** (1902), 1–53.

[3] F. H. Shu, F. C. Adams, S. Lizano, *Star formation in molecular clouds: observation and theory*, Ann. Rev. Astron. Astrophys. **25** (1987), 23–81.

[4] R. B. Larson, *Numerical calculations of the dynamics of a collapsing proto-star*, Mon. Not. R. Astron. Soc. **145** (1969), 271–295.

[5] M. V. Penston, *Dynamics of self-gravitating gaseous spheres*, Mon. Not. R. Astron. Soc. **144** (1969), 425–448.

[6] J. F. Alves, C. J. Lada, E. A. Lada, *Internal structure of a cold dark molecular cloud inferred from the extinction of background starlight*, Nature **409** (2001), 159–161.

[7] P. André, D. Ward-Thompson, M. Barsony, *From prestellar cores to protostars: the initial conditions of star formation*, Protostars and Planets IV (2000), 59–96.

[8] F. Motte, P. André, R. Neri, *The initial conditions of star formation in the Ophiuchus main cloud*, Astron. Astrophys. **336** (1998), 150–172.

[9] P. M. Solomon, A. R. Rivolo, J. Barrett, A. Yahil, *Mass, luminosity, and line width relations of Galactic molecular clouds*, Astrophys. J. **319** (1987), 730–741.

[10] W. B. Bonnor, *Boyle's law and gravitational instability*, Mon. Not. R. Astron. Soc. **116** (1956), 351–359.

[11] C. J. Lada et al., *Barnard 68: where did it come from?*, Astrophys. J. **586** (2003), 286–295.

[12] E. A. Bergin, M. Tafalla, *Cold dark clouds: the initial conditions for star formation*, Ann. Rev. Astron. Astrophys. **45** (2007), 339–396.

A Python Implementation

The following Python script reproduces all numerical results and figures.

Listing 1: Exact gravitational collapse and fragmentation via $\mathcal{F}_\lambda$.

```
#!/usr/bin/env python3
"""
flambda_astrophysics.py
Application de la classe F_lambda et de la hierarchie A_n
a l'effondrement gravitationnel des nuages de gaz interstellaires.
Modele : rho_dot = rho * exp(lambda * rho)
Formule centrale : T*_collapse = E1(lambda * rho0)
"""

import numpy as np
import matplotlib.pyplot as plt
from scipy.special import expi, sici
from scipy.optimize import brentq
from scipy.integrate import solve_ivp

GAMMA = 0.5772156649

plt.rcParams.update({
    'figure.dpi': 130,
    'axes.spines.top': False,
```

```
    'axes.spines.right': False,
    'font.size': 11
})

# ============================================================
# 1. Fonctions de base E1, En
# ============================================================

def E1(x):
    x = np.asarray(x, dtype=float)
    return np.where(x > 0, -expi(-x), np.inf)

def E1_scalar(x):
    if x <= 0: return np.inf
    return float(-expi(-x))

def E1_inv(w):
    if not np.isfinite(w) or w <= 0: return 1e-15
    lo, hi = 1e-12, 1e3
    while E1_scalar(lo) < w:
        lo /= 10
        if lo < 1e-300: return lo
    while E1_scalar(hi) > w:
        hi *= 10
    return brentq(lambda u: E1_scalar(u) - w, lo, hi, xtol=1e-12)

E1_inv_vec = np.vectorize(E1_inv)

def En_scalar(n, x):
    """E_n(x) par recursion."""
    if x <= 0: return np.inf
    if n == 1: return E1_scalar(x)
    e_prev = E1_scalar(x)
    for k in range(2, n+1):
        e_curr = (np.exp(-x) - x * e_prev) / (k - 1)
        e_prev = e_curr
    return e_curr

def T_star_An(n, lam, rho0):
    if lam <= 0 or rho0 <= 0: return np.inf
    return lam**(n-1) * En_scalar(n, lam * rho0)

def heat_semigroup_fft(W0, dx, t):
    N = len(W0)
    x_ker = np.fft.fftfreq(N, d=1.0/N) * dx
    kernel = np.exp(-x_ker**2 / (4*t)) / np.sqrt(4*np.pi*t)
    kernel *= dx
    return np.real(np.fft.ifft(np.fft.fft(W0) * np.fft.fft(kernel))
        )

# ============================================================
```

```
# 2. Objets astrophysiques calibres
#    rho = densite normalisee (rho_obs / rho_Jeans)
#    rho_Jeans ~ 10^-19 kg/m3 pour nuage moleculaire typique
#    T* = temps libre (free-fall time) normalise
#    Unites : T* en unites de t_ff = sqrt(3pi/32Grho0)
# ============================================================

# Densite de Jeans de reference
RHO_JEANS = 1.0   # normalise a 1

objets = {
    'Barnard 68': {
        'rho0':      0.20,    # 0.2 rho_Jeans (nuage pre-stellaire
           stable)
        'T_ff':      3.5,     # temps libre en unites t_ff
        'rho_peak':  8.0,
        'color': 'steelblue', 'marker': 'o', 'ms': 11,
        'description': 'Nuage de Bok, effondrement lent isotherme'
    },
    'Classe 0 protostellaire': {
        'rho0':      0.50,    # deja supercritique
        'T_ff':      1.8,
        'rho_peak':  50.0,
        'color': 'darkorange', 'marker': 's', 'ms': 11,
        'description': 'Proto toile jeune, collapse rapide'
    },
    'Coeur dense Ophiuchus': {
        'rho0':      0.80,    # tres supercritique
        'T_ff':      0.8,
        'rho_peak':  200.0,
        'color': 'seagreen', 'marker': '^', 'ms': 11,
        'description': 'Coeur pre-stellaire tres dense'
    },
    'Nuage geant moleculaire': {
        'rho0':      0.05,    # sous-critique
        'T_ff':      12.0,    # effondrement tres lent
        'rho_peak':  2.0,
        'color': 'crimson', 'marker': 'D', 'ms': 11,
        'description': 'GMC, effondrement sur echelle cosmique'
    }
}

print("Calibration : rho = densite/rho_Jeans, T* en unites t_ff")
print("="*72)
print(f"{'Objet':<30}  {'rho0':>6}  {'T*_obs':>8}  {'lambda':>10}  
   {'check':>8}")
print('-'*72)

for name, d in objets.items():
    lam = E1_inv(d['T_ff']) / d['rho0']
    d['lambda'] = lam
```

```
    T_check = E1_scalar(lam * d['rho0'])
    print(f"{name:<30}  {d['rho0']:>6.2f}  {d['T_ff']:>8.1f}  "
          f"{lam:>10.4f}  {T_check:>8.2f}")

# ============================================================
# 3. Figure 1 : trajectoires et prediction F_lambda
# ============================================================

print("\nFigure 1 : trajectoires d'effondrement et prediction F_lambda...")

fig, axes = plt.subplots(2, 2, figsize=(13, 8))
axes = axes.flatten()

for ax, (name, d) in zip(axes, objets.items()):
    lam  = d['lambda']
    rho0 = d['rho0']
    T_ob = d['T_ff']

    t_all = np.linspace(0, T_ob * 1.4, 300)
    rho_syn = np.where(
        t_all <= T_ob,
        rho0 * np.exp((np.log(d['rho_peak']/rho0) / T_ob) * t_all),
        d['rho_peak'] * np.exp(-0.5*(t_all - T_ob))
    )

    T_pred   = E1_scalar(lam * rho0)
    t_flam   = np.linspace(0, T_pred * 0.97, 400)
    rho_flam = np.array([E1_inv(E1_scalar(lam*rho0) - ti) / lam
                         for ti in t_flam])

    ax.plot(t_all,  rho_syn,  d['color'], lw=2, label='Donnees
       observees')
    ax.plot(t_flam, rho_flam, 'r--',      lw=2,
            label=f'F_lambda (lam={lam:.3f})')
    ax.axvline(T_pred, color='orange', lw=1.5, ls=':',
               label=f'T* = {T_pred:.1f} t_ff')
    ax.set_xlabel('t / t_ff'); ax.set_ylabel('rho / rho_Jeans')
    ax.set_title(f'{name}\n{d["description"]}', fontsize=9)
    ax.legend(fontsize=7)

plt.suptitle('Effondrement gravitationnel    Modele F_lambda
   calibre',
             fontsize=12)
plt.tight_layout()
plt.savefig('astrophysique_calibration.pdf', dpi=150)
plt.show()

# ============================================================
# 4. Figure 2 : hierarchie A_n     regimes physiques d'effondrement
#    A_1 : isotherme (T=cst, nuage froid)
```

```
#     A_2 : adiabatique (P ~ rho^gamma, gamma=5/3)
#     A_3 : rayonnement dominant (pression radiation)
#     A_4 : effondrement relativiste (noyau stellaire)
# ============================================================

print("Figure 2 : hierarchie A_n    regimes physiques...")

lam_range = np.logspace(-1.5, 1.5, 200)
rho0_fixed = objets['Classe 0 protostellaire']['rho0']

labels_n = [
    'A_1 : isotherme (T=cst)',
    'A_2 : adiabatique (gamma=5/3)',
    'A_3 : pression radiation',
    'A_4 : effondrement relativiste',
    'A_5 : singularite nue'
]
colors_n = ['steelblue', 'darkorange', 'seagreen', 'crimson', '
   purple']

fig, axes = plt.subplots(1, 2, figsize=(13, 5))

for n, col, lab in zip([1,2,3,4,5], colors_n, labels_n):
    T_vals = np.array([T_star_An(n, lam, rho0_fixed) for lam in
       lam_range])
    T_vals = np.clip(T_vals, 0, 10)
    axes[0].semilogx(lam_range, T_vals, color=col, lw=2, label=lab)

axes[0].axhline(0, color='black', lw=0.5)
axes[0].set_xlabel('lambda (rapport pression/gravite)')
axes[0].set_ylabel('T*_n (t_ff)')
axes[0].set_title(f'Hierarchie A_n    T* vs lambda\n'
                  f'(Classe 0, rho0={rho0_fixed})')
axes[0].legend(fontsize=7)
axes[0].set_ylim(-0.3, 10)

# Regimes asymptotiques lambda->0
lam_small = np.logspace(-1.5, -0.2, 100)
T_A1 = -np.log(lam_small) - np.log(rho0_fixed) - GAMMA
T_A2 = lam_small
T_A3 = lam_small**2 / 2

axes[1].semilogx(lam_small, np.clip(T_A1, 0, 8), 'steelblue', lw=2,
                 label='A_1 : -ln(lambda) [log, isotherme]')
axes[1].semilogx(lam_small, T_A2, 'darkorange', lw=2,
                 label='A_2 : lambda [lineaire, adiabatique]')
axes[1].semilogx(lam_small, T_A3, 'seagreen', lw=2,
                 label='A_3 : lambda^2/2 [quadratique, radiation]')
axes[1].set_xlabel('lambda')
axes[1].set_ylabel('T* asymptotique (t_ff)')
axes[1].set_title('Trois regimes de bifurcation\n(Theoreme 6,
```

```
    Brindel 2026)')
axes[1].legend(fontsize=8)
axes[1].set_ylim(0, 6)

plt.tight_layout()
plt.savefig('hierarchie_An_astro.pdf', dpi=150)
plt.show()

# ============================================================
# 5. Figure 3 : comparaison F_lambda vs modele de Jeans classique
#    Jeans : instabilite si M > M_Jeans, temps libre t_ff = sqrt(3 pi/32Grho)
#    F_lambda : T* = E1(lambda * rho0) depend explicitement de rho0 et lambda
# ============================================================

print("Figure 3 : F_lambda vs modele de Jeans classique...")

# Temps de free-fall Jeans (normalise) : t_ff = 1/sqrt(G*rho)
# En unites normalisees : T_Jeans = 1.0 (constant, independant de rho0)
T_Jeans_norm = 1.0

rho0_range = np.linspace(0.05, 1.5, 100)

# F_lambda pour classe 0
lam_c0 = objets['Classe 0 protostellaire']['lambda']
T_Flambda_c0 = E1(lam_c0 * rho0_range)

# F_lambda pour Barnard 68
lam_b68 = objets['Barnard 68']['lambda']
T_Flambda_b68 = E1(lam_b68 * rho0_range)

fig, axes = plt.subplots(1, 2, figsize=(13, 5))

axes[0].axhline(T_Jeans_norm, color='gray', lw=2, ls='--',
                label='Jeans classique (t_ff = cst)')
axes[0].plot(rho0_range, T_Flambda_c0, 'darkorange', lw=2.5,
             label=f'F_lambda Classe 0 (lam={lam_c0:.3f})')
axes[0].plot(rho0_range, T_Flambda_b68, 'steelblue', lw=2.5,
             label=f'F_lambda Barnard 68 (lam={lam_b68:.3f})')

for name, d in objets.items():
    axes[0].plot(d['rho0'], E1_scalar(d['lambda']*d['rho0']),
                 marker=d['marker'], color=d['color'],
                 ms=d['ms'], zorder=6)

axes[0].set_xlabel('rho0 / rho_Jeans')
axes[0].set_ylabel('T*_collapse (t_ff)')
axes[0].set_title('F_lambda vs Jeans classique\n'
                  '(Jeans : T independant de rho0)')
```

```
axes[0].legend(fontsize=9)
axes[0].set_ylim(0, 6)

# Surface T* = E1(lambda * rho0) dans l'espace (lambda, rho0)
lam_g = np.linspace(0.01, 5.0, 150)
rho_g = np.linspace(0.01, 1.5, 150)
LAM, RHO = np.meshgrid(lam_g, rho_g)
T_map = np.clip(E1(LAM * RHO), 0, 8)

im = axes[1].contourf(LAM, RHO, T_map, levels=20, cmap='RdYlGn_r')
axes[1].contour(LAM, RHO, T_map, levels=[0.5, 1, 2, 4],
                colors='white', linewidths=0.8, linestyles='--')
plt.colorbar(im, ax=axes[1], label='T*_collapse (t_ff)')

for name, d in objets.items():
    axes[1].plot(d['lambda'], d['rho0'],
                 marker=d['marker'], color=d['color'],
                 ms=d['ms'], zorder=6, label=name.split(' ')[0])

axes[1].set_xlabel('lambda'); axes[1].set_ylabel('rho0 / rho_Jeans')
axes[1].set_title('Surface T* = E1(lambda * rho0)')
axes[1].legend(fontsize=7, loc='upper right')

plt.tight_layout()
plt.savefig('jeans_vs_flambda.pdf', dpi=150)
plt.show()

# ============================================================
# 6. Figure 4 : diagramme de phase effondrement / stabilite
# ============================================================

print("Figure 4 : diagramme de phase effondrement / stabilite...")

T_horizon = 2.0   # 2 t_ff = horizon de formation stellaire
lam_g2 = np.linspace(0.01, 6.0, 300)
rho_g2 = np.linspace(0.01, 1.5, 300)
LAM2, RHO2 = np.meshgrid(lam_g2, rho_g2)
collapse_map = (E1(LAM2 * RHO2) < T_horizon).astype(float)
boundary     = np.array([E1_inv(T_horizon) / li for li in lam_g2])
boundary     = np.clip(boundary, 0.01, 1.5)

fig, axes = plt.subplots(1, 2, figsize=(13, 5))

axes[0].contourf(LAM2, RHO2, collapse_map,
                 levels=[-0.5, 0.5, 1.5],
                 colors=['#c8e6c9', '#ffcdd2'], alpha=0.85)
axes[0].plot(lam_g2, boundary, 'k-', lw=2.5,
             label=f'Frontiere T* = {T_horizon} t_ff')

for name, d in objets.items():
```

```
    axes[0].plot(d['lambda'], d['rho0'],
                 marker=d['marker'], color=d['color'],
                 ms=d['ms'], zorder=6,
                 label=name.split(' ')[0])

axes[0].text(3.5, 1.2, 'COLLAPSE\n(<2 t_ff)', fontsize=10,
             color='#c62828', fontweight='bold')
axes[0].text(0.2, 0.3, 'STABLE\n(>2 t_ff)', fontsize=10,
             color='#2e7d32')
axes[0].set_xlabel('lambda (feedback gravitationnel)')
axes[0].set_ylabel('rho0 / rho_Jeans')
axes[0].set_title(f'Diagramme de phase    horizon {T_horizon} t_ff')
axes[0].legend(fontsize=7, loc='upper right')

# Loi universelle T* ~ -ln(lambda)
lam_range2 = np.logspace(-1, 1, 200)
rho0_vals  = [0.2, 0.5, 0.8]
cols_u     = ['steelblue', 'darkorange', 'seagreen']

for r0, col in zip(rho0_vals, cols_u):
    T_ex   = E1(lam_range2 * r0)
    T_asym = -np.log(lam_range2) - np.log(r0) - GAMMA
    axes[1].semilogx(lam_range2, np.clip(T_ex, 0, 8),
                     color=col, lw=2, label=f'rho0={r0}')
    axes[1].semilogx(lam_range2, np.clip(T_asym, 0, 8),
                     color=col, lw=1.2, ls='--')

axes[1].set_xlabel('lambda')
axes[1].set_ylabel('T*_collapse (t_ff)')
axes[1].set_title('Loi universelle T* ~ -ln(lambda)\n'
                  '(plein=exact , tiret=asymptotique)')
axes[1].legend(fontsize=9)
axes[1].set_ylim(0, 8)

plt.tight_layout()
plt.savefig('phase_collapse.pdf', dpi=150)
plt.show()

# ============================================================
# 7. Figure 5 : fragmentation spatiale du nuage via PDE
#    x = coordonnee radiale du nuage
#    U(x,t) = densite locale
#    Fragmentation = plusieurs pics de densite -> plusieurs etoiles
# ============================================================

print("Figure 5 : fragmentation spatiale du nuage via PDE F_lambda ...")

L_cloud  = 10.0
N_cloud  = 512
```

```
x_cloud  = np.linspace(-L_cloud, L_cloud, N_cloud, endpoint=False)
dx_cloud = x_cloud[1] - x_cloud[0]
lam_cloud = objets['Barnard 68']['lambda']

# Nuage avec perturbation initiale : pic central + sous-structures
rho_init = (0.15 * np.exp(-x_cloud**2 / 2.0) +
            0.08 * np.exp(-(x_cloud-3)**2 / 0.5) +
            0.06 * np.exp(-(x_cloud+3)**2 / 0.5))
rho_init = np.clip(rho_init, 1e-6, None)
W0_cloud = E1(lam_cloud * rho_init)

plt.figure(figsize=(10, 5))
cmap5 = plt.cm.hot
t_frag = [0, 0.5, 1.0, 2.0, 3.0, 5.0]

for i, ti in enumerate(t_frag):
    if ti == 0:
        U_t = rho_init.copy()
    else:
        W_t = heat_semigroup_fft(W0_cloud, dx_cloud, ti) + ti
        U_t = E1_inv_vec(W_t) / lam_cloud
    plt.plot(x_cloud, U_t, color=cmap5(0.15 + 0.7*i/len(t_frag)),
             lw=1.8, label=f't={ti} t_ff')

plt.xlabel('r / r_Jeans')
plt.ylabel('rho / rho_Jeans')
plt.title('Fragmentation spatiale du nuage   PDE F_lambda\n'
          'Plusieurs pics = formation stellaire multiple (Thm 14)')
plt.xlim(-8, 8); plt.ylim(bottom=0)
plt.legend(fontsize=8, ncol=3)
plt.tight_layout()
plt.savefig('fragmentation_nuage.pdf', dpi=150)
plt.show()

# ============================================================
# 8. Figure 6 : oscillations de Bonnor-Ebert et invariant Omega
#    Nuages oscillants : lambda = i*mu (regime complexe)
#    Omega = pi/2 - Si(mu * rho0) conserve exactement
#    Interpretation : compteur de periodes d'oscillation
# ============================================================

print("Figure 6 : oscillations de Bonnor-Ebert et invariant Omega ...")

mu_be = objets['Barnard 68']['lambda']

# Oscillations de Bonnor-Ebert : densite oscillante amortie
t_osc   = np.linspace(0, 30, 2000)
rho_osc = 0.25 * (1 + 0.6*np.sin(2*np.pi*t_osc/4.0)) * \
          np.exp(-0.03*t_osc)
rho_osc = np.clip(rho_osc, 1e-6, None)
```

```
413
414 Si_be, _ = sici(mu_be * rho_osc)
415 Omega_be  = np.pi/2 - Si_be
416
417 fig, axes = plt.subplots(3, 1, figsize=(11, 9), sharex=True)
418
419 axes[0].plot(t_osc, rho_osc, 'steelblue', lw=1.5)
420 axes[0].set_ylabel('rho / rho_Jeans')
421 axes[0].set_title('Oscillations de Bonnor-Ebert (nuage pre-
    stellaire)')
422
423 axes[1].plot(t_osc, Omega_be, 'darkorange', lw=1.5)
424 axes[1].set_ylabel('Omega = pi/2 - Si(mu*rho)')
425 axes[1].set_title('Invariant topologique Omega (Theoreme 8, Brindel
    2026)')
426
427 axes[2].plot(t_osc, np.gradient(Omega_be, t_osc), 'seagreen', lw
    =1.2)
428 axes[2].axhline(0, color='black', lw=0.5)
429 axes[2].set_ylabel('dOmega/dt')
430 axes[2].set_xlabel('t / t_ff')
431 axes[2].set_title('Derivee de Omega    pics = passages par rho_max
    ')
432
433 plt.tight_layout()
434 plt.savefig('oscillations_bonnor_ebert.pdf', dpi=150)
435 plt.show()
436
437 # ============================================================
438 # 9. Figure 7 : signal d'alerte pour observations radio
439 #    T*_restant(t) = E1(lambda * rho(t))
440 #    Observable depuis mesures de densite de colonne (telescopes
    radio)
441 # ============================================================
442
443 print("Figure 7 : signal d'alerte pour observations radio...")
444
445 d_c0   = objets['Classe 0 protostellaire']
446 lam_c0 = d_c0['lambda']
447 T_ob   = d_c0['T_ff']
448 tau    = np.linspace(0, T_ob * 0.95, 200)
449
450 rho_traj  = d_c0['rho0'] * np.exp(
451     (np.log(d_c0['rho_peak'] / d_c0['rho0']) / T_ob) * tau)
452 T_restant = E1(lam_c0 * rho_traj)
453
454 fig, axes = plt.subplots(2, 1, figsize=(10, 6), sharex=True)
455
456 axes[0].plot(tau, rho_traj, 'darkorange', lw=2)
457 axes[0].set_ylabel('rho / rho_Jeans')
458 axes[0].set_title('Classe 0 protostellaire    Signal d\'alerte')
```

```
459
460 axes[1].plot(tau, T_restant, 'steelblue', lw=2,
461              label='T*_restant = E1(lambda * rho(t))')
462 axes[1].axhline(0.2, color='red', lw=1.5, ls='--',
463                 label='Seuil critique : 0.2 t_ff')
464 axes[1].fill_between(tau, 0, T_restant,
465                      where=(T_restant < 0.2),
466                      color='red', alpha=0.3,
467                      label='Formation stellaire imminente')
468 axes[1].set_xlabel('t / t_ff')
469 axes[1].set_ylabel('T* restant (t_ff)')
470 axes[1].legend(fontsize=9)
471 axes[1].set_ylim(bottom=0)
472
473 plt.tight_layout()
474 plt.savefig('alerte_protostellaire.pdf', dpi=150)
475 plt.show()
476
477 # ============================================================
478 # 10. Tableau hierarchie A_n pour chaque objet
479 # ============================================================
480
481 print("\n" + "="*72)
482 print("HIERARCHIE A_n    T*_n pour chaque objet astrophysique")
483 print("="*72)
484
485 regimes = {1:'-ln(lam) isotherme', 2:'lam adiabatique',
486            3:'lam^2/2 radiation', 4:'lam^3/6 relativiste',
487            5:'lam^4/24 singularite'}
488
489 for name, d in objets.items():
490     lam_i = d['lambda']
491     rho_i = d['rho0']
492     print(f"\n{name} (lambda={lam_i:.4f}, rho0={rho_i:.2f})")
493     print(f"  {'n':>3}  {'T*_n (t_ff)':>14}  {'Regime':>22}")
494     print(f"  {'-'*44}")
495     for n in range(1, 6):
496         T_n = T_star_An(n, lam_i, rho_i)
497         print(f"  {n:>3}  {T_n:>14.4f}  {regimes[n]:>22}")
498
499 # ============================================================
500 # 11. Recapitulatif
501 # ============================================================
502
503 print("\n" + "="*72)
504 print("R CAPITULATIF    Classe F_lambda en astrophysique")
505 print("="*72)
506 print(f"\n{'Objet':<30}  {'lambda':>8}  {'rho0':>6}  "
507       f"{'T*_exact':>10}  {'T*_obs':>8}")
508 print('-'*70)
509 for name, d in objets.items():
```

```
    T_ex = E1_scalar(d['lambda'] * d['rho0'])
    print(f"{name:<30}  {d['lambda']:>8.4f}  {d['rho0']:>6.2f}  "
          f"{T_ex:>10.2f}  {d['T_ff']:>8.1f}")

print("""
Formules cles :
  Calibration       = lambda = E1_inv(T*_obs) / rho0
  T*_collapse       = E1(lambda * rho0)          [unites t_ff]
  Hierarchie A_n    = T*_n = lambda^{n-1} * En(lambda * rho0)
  Loi universelle   = T* ~ -ln(lambda)           [A_1, lambda->0]
  Fragmentation     = E1(lambda U(x,t)) = e^{tDelta}W0 + t
  Signal alerte     = T*_restant(t) = E1(lambda * rho(t))
  Invariant Omega   = pi/2 - Si(mu * rho0)       [oscillations BE]

Regimes physiques A_n :
  n=1 : effondrement isotherme (T=cst)
  n=2 : effondrement adiabatique (gamma=5/3)
  n=3 : pression de radiation dominante
  n=4 : effondrement relativiste (noyau stellaire)
  n=5 : singularite nue (conjecture censure cosmique)

Figures sauvegardees :
  astrophysique_calibration, hierarchie_An_astro, jeans_vs_flambda,
  phase_collapse, fragmentation_nuage, oscillations_bonnor_ebert,
  alerte_protostellaire
""")
print("   Simulation terminee.")
```